本文 关于
1.对付selfish的直接评价-- 统计抽样理论 估计出合理的全局值
2.timeout的选取 采用 2005G的延迟 计算结果
3.benchmark 2013adhocnet & 2015IngRChen
4.通信能量的计算

important paper!!!!

Journal of Network and Computer Applications 109 (2018) 11–23

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Wise-PRoPHET: A Watchdog supervised PRoPHET for reliable dissemination of post disaster situational information over smartphone based DTN

Souvik Basu [a,*], Ayanesh Biswas [a], Siuli Roy [a], Sipra DasBit [b]

[a] Computer Application Centre, Heritage Institute of Technology, Kolkata, India
[b] Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur, India

ARTICLE INFO

Keywords:
Delay tolerant network
Post disaster communication network
PRoPHET
Watchdog
Trust
Statistical estimation
Markov chain

基于watchdog技术的trust

ABSTRACT

Delay tolerant network (DTN) has been successfully proposed for setting up emergency post disaster communication networks when normal communication infrastructure is typically incapacitated. These networks work on the basis of cooperation from participating nodes, which is cost-intensive in terms of battery life, computation, etc. Therefore, nodes can refuse to cooperate to save resources, giving rise to selfish nodes that hinder the transmission of sensitive post disaster situational messages. Another issue is the presence of malicious nodes that collude to either spoil the reputation of honest nodes or boost the reputation of selfish nodes. The existing DTN routing protocols, like PRoPHET, do not address these issues. In this paper, a trust based Watchdog technique is seamlessly integrated with PRoPHET so that situational messages are successfully delivered even in the presence of selfish and malicious nodes. The Watchdog monitors its neighbouring nodes to generate a local perception about their forwarding behaviour. This information is then disseminated in the network to build a global perception of forwarding behaviour for detection of selfish nodes. The local perception is further used to identify malicious nodes in the network. The proposed technique rationalizes self-trusting, a property of trust based data forwarding in opportunistic networks which reduces and delays message transfers, to further improve delivery ratio and delay. Results of extensive simulation, using ONE simulator, substantiate the efficiency of the proposed Watchdog enabled PRoPHET over state-of-the-art competing schemes, in terms of detection ratios, attraction ratio, etc. while not compromising standard network performance. Finally, it is claimed that the proposed technique, tolerates a reasonable percentage of selfish and malicious nodes to achieve a desirable level of network performance, in a post disaster communication scenario.

1. Introduction

Disaster management is essentially information management because facilitating the access, exchange and diffusion of reliable situational information are most important for effective decision making and risk reduction (International Federation of Red Cross and Red Crescent Societies, 2013). However, owing to the fact that, cellular and other traditional communication facilities become non-functional during disasters (Luo et al., 2010), it becomes difficult to collect and exchange situational information from remote and inaccessible shelters. The networking research community has strongly proposed the use of delay tolerant network (DTN) for setting up an emergency post disaster communication network (Fall et al., 2010; Chenji et al., 2011; Ntareme et al., 2011; Campillo et al., 2013). The DTN offers store-carry-forward protocols that enable communication during impaired connectivity, although with certain delay (Jain et al., 2005). Increasing availability of smartphones and their integration with technologies like Bluetooth, WiFi direct, etc. can be effectively harnessed to form a DTN during or after a disaster. Volunteers and relief workers carrying such devices working in the DTN mode can opportunistically exchange situational information.

Flooding-based DTN routing protocols, like Epidemic (Vahdat and Becker, 2000), forward many replicas of the same message and consumes much resources; making them inappropriate for resource constrained smartphone based DTN. Probability-based routings, like PRoPHET (Lindgren et al., 2003) and MaxProp (Burgess et al., 2006), forward a message to a node having the highest probability of encountering the

* Corresponding author.
E-mail address: souvik.basu@heritageit.edu (S. Basu).

https://doi.org/10.1016/j.jnca.2018.02.012
Received 28 April 2017; Received in revised form 21 September 2017; Accepted 22 February 2018
Available online 28 February 2018

destination. Therefore, the success of such routing schemes depends on the cooperation from participating nodes, which, however, is quite cost intensive for these nodes. Thus, one source of problem in DTNs is the presence of selfish nodes who refuse to forward messages for others, to save their own resources. Selfish nodes, by arbitrarily dropping packets, can seriously degrade transmission of sensitive situational messages to the control station. Another source of problem in DTNs is the presence of malicious nodes that indulge in either - bad-mouthing (spreading low trust values for cooperating nodes) or ballot-stuffing (spreading high trust values for selfish nodes) (Chen et al., 2013). This disrupts the correct behaviour of the network as a large number of messages are directed towards selfish nodes and are eventually dropped. Therefore, detecting and avoiding such selfish and malicious nodes while routing messages is crucial for effective disaster management operations. However, the existing PRoPHET routing protocol does not address these issues.

The Watchdog technique has been extensively used to detect selfish and misbehaving nodes in MANETs and WSNs ((Orallo et al., 2015) - (Cho et al., 2012)). However, these techniques either rely on the assumption of the relatively slow mobility of nodes or depend on contemporary end-to-end routing paths between the source and the destination. None of these Watchdog techniques work for our DTN based smartphone network because neither existence of contemporary end-to-end routing paths nor relatively slow network mobility can be assumed in a DTN scenario.

In this paper, a trust based Watchdog technique for DTNs is proposed where a Watchdog enabled node (smartphone carried by volunteer/ relief worker) monitors its neighbouring nodes to generate a local perception about their forwarding behaviour - selfish or altruistic. This knowledge is then dispersed in the network to build a global perception about their forwarding behaviour of all nodes in the network. The Watchdog is further enhanced to identify malicious nodes and stop using incorrect trust information provided by them. Finally, we intertwine the Watchdog technique with PRoPHET for enumerating the ability of a node to deliver messages not only based on its probability of encountering the destination but also depending on its trustworthiness as an honest forwarder. The proposed Wise-PRoPHET technique successfully detects and avoids selfish and malicious nodes and routes sensitive post disaster situational messages through the best possible forwarders.

The rest of this paper is organized as follows. Section 2, summarizes related work in this field. In Section 3, we describe the system model related to our work. We present the Wise-PRoPHET technique in Section 4. Performance of the Wise-PRoPHET technique is evaluated both quantitatively and qualitatively in Section 5. We conclude the paper with a direction towards future work in Section 6.

2. Related work

In this section, we first review works on trust based Watchdog techniques for MANETs and WSNs. Subsequently, existing researches on trust management systems for DTNs are reported.

2.1. Trust based systems for MANETs and WSNs

Substantial researches have been conducted on trust based Watchdog techniques for combating forwarding misbehavior in MANETs (Orallo et al., 2015; Marti et al., 2000; Wang et al., 2014; Marmol and Perez, 2012) and WSNs (Zhou et al., 2015; Cho et al., 2012). These research, aimed at detecting malicious nodes, can be broadly classified into two groups – the first group relies on channel monitoring and the other depends on destination acknowledgment. In the first group of protocols, if the message sent out from the next-hop forwarder is exactly the same as the original one sent out from the previous-hop node (i.e., the Watchdog owner) the next-hop forwarder is supposed to be a benign node behaving well for data forwarding. Successful channel monitoring, however, relies on the assumption of relatively slow mobility of nodes. In the second group of protocols, acknowledgments are generated at the destination when data are received, and then they are routed back to the data source along the same transmission path. Acknowledgements demonstrate that data are successfully delivered to the destination. These acknowledgments, however, can be received by the source only if contemporary routing paths exist between the source and the destination node.

As stated before, assumptions like relatively slow mobility and the existence of contemporary routing paths do not apply for our post disaster communication network that has DTN as its underlying architecture. DTNs are characterized by unpredictable mobility patterns, frequent network partitioning and other issues (Zhu et al., 2014).

2.2. Trust based systems for DTNs

In recent years there have been quite a few works on misbehavior detection in DTNs, most of which are either trust based or incentive based. Some of the trust based approaches are reported here.

Li et al. in (Li and Das, 2013) propose a novel trust-based framework which can be flexibly integrated with a large family of existing single-copy data forwarding protocols in OppNets, aiming at providing comprehensive evaluation to an encounter's ability to deliver data. With the help of the proposed framework, black hole and arbitrary forwarding attacks can be counteracted effectively. Authors propose a Positive Feedback Message (PFM) as the evidence of the forwarding behaviour of a node, which is fed into the Watchdog component in the framework.

Zhu et al. in (Zhu et al., 2014) propose iTrust, a probabilistic misbehavior detection scheme, for secure DTN routing towards efficient trust establishment. The basic idea of iTrust is introducing a periodically available Trusted Authority (TA) to judge the node's behaviour based on the collected routing evidences and probabilistically checking. Authors model iTrust as an Inspection Game and use game theoretical analysis to demonstrate that by setting an appropriate investigation probability, TA could ensure the security of DTN routing at a reduced cost.

A trust based cooperative watchdog system is proposed for Vehicular DTNs by Dias et al. in (Dias et al., 2015) to detect and act against misbehaving nodes in order to reduce their impact on the overall network performance. Its operation relies on a cooperative exchange of nodes reputation along the network. By detecting selfish or misbehaving nodes, it is possible to improve the overall network performance.

Chen et al. in (Chen et al., 2013) and (Chen et al., 2011) design and validate a trust management protocol for DTNs and apply it to secure routing to demonstrate its utility. The protocol combines QoS trust with social trust to obtain a composite trust metric. Results obtained at the design time facilitate dynamic trust management for DTN routing in response to dynamically changing conditions at runtime.

Dini et al. in (Dini and Duca, 2012) propose a reputation-based protocol for contrasting blackholes. Every node locally maintains the reputation of forwarding nodes it comes in touch with and, then, upon selecting the next forwarding node, the node chooses among those having the highest reputation. The proposed reputation protocol is composed of three basic mechanisms—acknowledgments, node lists, and aging. These mechanisms make the communication more efficient and capable of adapting to the changing operating conditions of a DTN.

Finally, Ayday et al. in (Ayday et al., 2010) introduce a robust and efficient security mechanism for DTNs. This consists of a trust management mechanism and an iterative trust and reputation mechanism (ITRM). The trust management mechanism enables each node to determine the trustworthiness of the nodes that it had a direct transaction. On the other hand, ITRM takes advantage of an iterative mechanism to detect and isolate the malicious nodes from the network in a short time.

From the above study, it is evident that extensive researches have been conducted on trust based systems in DTN. However, the potential of integrating such a system with DTN routing protocols like, PRoPHET have seldom been explored. The proposed Wise-PRoPHET technique addresses certain fundamental issues relating to DTN trust that are hitherto not handled in any of the above works. The key contributions of this work are:

- All the above works compute trust/ reputation of a forwarder based on instantaneous feedbacks provided by a group of local nodes in the network. Although such trust values may generate a local perception about the forwarding behaviour of a node, they do not depict a global perception about the forwarding behaviour of the node. Wise-PRoPHET, using statistical estimation, generates and uses globally endorsed trust values rather than the locally perceived ones, resulting in more accurate selfishness detection.
- None of these above schemes attempt to enumerate the time required for the trust values to get propagated in the network and evaluate the timeliness of the proposed trust protocol. The proposed Wise-PRoPHET technique, using an absorbing finite-state Markov chain, determines expected message delays. This assists in enumerating the time required for the trust values to get disseminated in the network.
- The works that integrate a trust framework with routing protocols like PRoPHET, do not solve the *self-trusting* issue (Li and Das, 2013), a property of trust based data forwarding in opportunistic networks. Self-trusting reduces delivery ratio and increases the delay in delivering data to the destination. In this work, the trust based Watchdog technique is seamlessly integrated with PRoPHET avoiding self-trusting. This increases delivery ratio and reduces the delivery delay to a great extent.

Other major highlights of the proposed work are:

- The proposed technique ignores false trust values provided by colluding nodes while enumerating the forwarding behaviour of a node.
- A thorough overhead analysis in terms of computation and communication requirements is done to gauge the suitability of our proposed technique to work with the resource constrained low-power smartphones based DTN.
- Finally, the proposed technique is evaluated in ONE (Keranen et al., 2009) simulator using real disaster data of the 2015 Nepal earthquake.

3. System model

Our system model consists of network architecture, routing protocol, and adversary & trust models.

3.1. Network architecture

During and after a disaster, victims take shelter in nearby shelters and relief operations in these shelters are controlled from a central control station. In our work, we assume that the smartphones carried by the volunteers and relief workers communicate over a DTN and create a post disaster communication network. Such a DTN based post disaster communication network comprises of three types of nodes - shelter-node, control-node and forwarder-node (Basu et al., 2015). A typical post disaster communication network is illustrated in Fig. 1.

(i) *Shelter-node* - Each shelter consists of a shelter-node (e.g., a laptop or a workstation) that generates situational messages stating requirements in that shelter. The shelter-node periodically broadcasts these messages towards the forwarder-nodes that are working in and around it.

(ii) *Control-node* - The central control station consists of control-node (e.g., a laptop or a workstation) that receives situational information about the shelters, assesses their requirements and distributes relief stock to the shelters.

(iii) *Forwarder-node* - Volunteers carrying smartphones are forwarder-nodes that move across the disaster stricken area, exchange situational messages of the shelters and eventually transmit them to the control station. Such exchanges are accomplished using Bluetooth or WiFi-Direct interfaces of the smartphones. All disaster management instructions including commands and alerts from the control station are communicated to the different shelters through these nodes. A typical post disaster communication network is illustrated in Fig. 1.

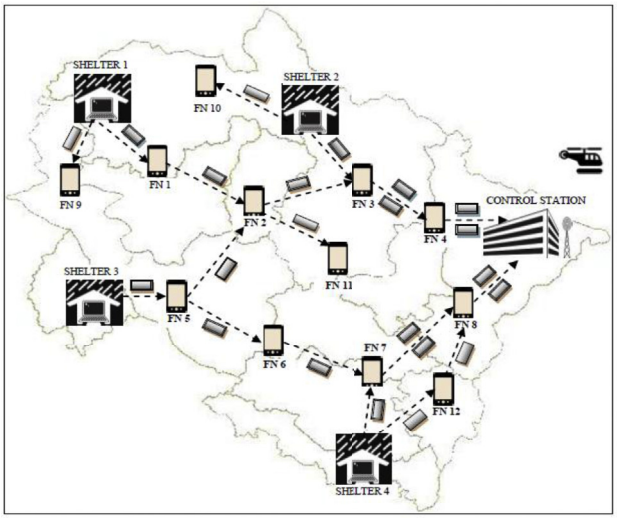

Fig. 1. A post disaster communication network.

3.2. Node and message ID generation

The proposed technique requires that each node (shelter-nodes, forwarder-nodes, and the control-node) and each message is assigned a unique ID. But, synchronizing these IDs in a DTN environment, without end-to-end communication among the nodes, becomes challenging.

In our proposed technique, the central control station, in the setup phase, assigns unique IDs to each shelter-node ($S_1, S_2, \ldots, S_p$), each forwarder-node ($FN_1, FN_2, \ldots, FN_N$) and the control-node (CN). If a new shelter is setup, the shelter-node therein has to register with the control station and get a shelter-id. Similarly, if a new volunteer wants to join the relief operation he/she has to register with the control station and get a forwarder-id. Thus, new entrants can be always accommodated into the network. Once these IDs are generated, the central control station has no role in the runtime phase when forwarder-nodes exchange situational messages of the shelters using the Wise-PRoPHET technique.

Each message is assigned a message ID at the time of its generation. Message ID is essentially a unique message number prefixed with the source ID, where source ID is the ID of the node (may be shelter-node or forwarder-node or control-node) that generates the message.

3.3. Routing protocol

The forwarder-nodes exchange messages among themselves following the DTN routing protocol PRoPHET (Lindgren et al., 2003). This protocol uses the history of previous encounters and the transitive property to estimate a delivery predictability (DP), $P(A,B)$, at each node A for all known destinations B, kept in a DP table. As nodes meet, they exchange DP tables and update their own DP table according to the following equations.

For an encountered node B, node A updates its DP using the following equation

$$P(A,B)_{new} = P(A,B)_{old} + \left(1 - P(A,B)_{old}\right) \times P_{enc}$$

where P_{enc} is a configurable parameter. Due to the transitive property of DP, A updates DPs for all other destinations i known by B based on the DP table provided by B. Transitive update is done using the following equation where β is a configurable parameter

带着信息走

$$P(A,i)_{new} = max\left(P(A,i)_{old}, P(B,i) \times P(A,B)_{new} \times \beta\right)$$

In order to eliminate stale information from the network, the *DP* table is periodically aged according to the following equation for all destinations *i*, where γ is a constant and *T* is the number of time units since the last aging process.

$$P(A,i)_{new} = P(A,i)_{old} \times \gamma^{T}$$

The message forwarding strategy in PRoPHET states – when a node *A* meets another node *B*, a message is transferred to *B* if the *DP* of the destination of the message is higher at *B*.

3.4. Adversary model

Although there can be several types of misbehaving nodes that may pose serious security threats to a DTN, we consider selfish and malicious nodes. We explain below the attacks launched by these adversaries:

3.4.1. Selfish nodes

A selfish node is one that uses the routing services but does not spend its own resources to cooperate towards that service (Dini and Duca, 2012). Such nodes drop all messages (destined to other nodes) forwarded to it, impairing message delivery to a great extent.

3.4.2. Malicious nodes

We define malicious nodes as a group of nodes that collude to launch attacks like - bad-mouthing (ruining the reputation of well-behaved nodes) and ballot-stuffing (boosting the reputation of selfish nodes). Malicious nodes set the trust value of all selfish nodes to 1 and that of all altruistic nodes to 0. Detection of selfish nodes becomes extremely challenging in presence of such malicious nodes.

3.5. Trust model

We define a forwarder-node's trust level as a real number in the range of [0, 1], with 1 indicating complete trust (fully altruistic), 0.5 ignorance, and 0 complete distrust (fully selfish). The various types of trusts used in our work are defined below:

3.5.1. Direct trust

Direct trust about a forwarder-node is the first-hand information about the node's forwarding behaviour. Such trust values are provided by the Watchdog that directly monitors the node. Direct trust is subjective (perception about a node's trustworthiness vary from node to node) and asymmetric (two nodes may not have similar mutual trust)

3.5.2. Indirect trust

Direct trust value gathered by a node, about a specific node, is shared with other nodes in the network. Other nodes use this second-hand information as *indirect trust* about that specific node. Such indirect trusts or recommendations help in the fast diffusion of trust information across the network and are used to derive four types of trust values. *Local Trust* about a forwarder-node is the average of direct trust values about that node, provided by a group of nodes, depicting a local perception about the forwarding behaviour of the node. *Global trust* about a forwarder-node is the average of direct trust values about that node, provided by all other nodes in the network, providing global perception about the forwarding behaviour of that node. *Forwarder trust* about a forwarder-node is the weighted average of direct and global trusts. *Rater trust* about a forwarder-node is the conformance of the direct trust values it provides (about other nodes) with the average direct trust value of these nodes.

4. Proposed Wise-PRoPHET technique

We propose the Wise-PRoPHET technique where a node selects the next hop forwarder based on the forwarder's probability of encountering the destination and its trustworthiness as an honest forwarder. The technique consists of three modules - Watchdog Operation, Indirect Trust Enumeration and Forwarder Selection.

4.1. Watchdog Operation

This module consists of two phases. In the first phase, a node, using the installed Watchdog, monitors the forwarding behaviour of neighbouring nodes. In the second phase, the node, based on the monitored behaviour, computes direct trust values of the neighbouring nodes.

4.1.1. Monitoring forwarding behaviour

When forwarder-node FN_i meets another node FN_j, FN_i monitors FN_j on the basis of a Forwarding Evidence (*FE*) sent by FN_k. FN_k is the node to which FN_j has forwarded the message. The utility of the *FE* is to convince FN_i that FN_j has actually forwarded the message to a competent forwarder FN_k.

Fig. 2 illustrates the mechanism. We explain the *FE* creation, propagation and quantification process below. For the FE creation and propagation we partially adopt the approach proposed in (Li and Das, 2013). However, our process of *FE* quantification is different from (Li and Das, 2013); instead of classifying them as only 'good' and 'bad', we objectively quantify the evidences for computing direct trust values.

等待 FEij

4.1.1.1. FE creation. To start with, when FN_i meets FN_j at time *t*, FN_i checks the delivery predictability of FN_j (refer to Section 3.3) and sends a message to it. After sending the message, FN_i waits for a *timeout* period for the arrival of the forwarding evidence $FE_{i,j}(t)$ pertaining to the message. *timeout* is the expected time required for the evidence to reach FN_i. A rational *timeout* period is derived in Section 4.4.3. FN_j carries the message until it meets a more competent node, FN_k, and sends it to FN_k. On receiving the message, FN_k creates $FE_{i,j}(t)$ destined to FN_i. $FE_{i,j}(t)$ consists of 12 attributes, the first 5 of which are fixed while the remaining 7 are updated at each hop of data forwarding. FN_k derives the attributes of $FE_{i,j}(t)$ from the message being received. The 12-tuple *FE* structure is shown in Fig. 3.

4.1.1.2. FE propagation. The $FE_{i,j}(t)$ created by FN_k, destined to FN_i, is transmitted over the network using epidemic routing (Vahdat and Becker, 2000). $FE_{i,j}(t)$ is required to be fast propagated in the network so that it quickly reaches FN_i in multiple hops. Therefore, there is no other alternative than exploiting the high delivery ratio property of a flooding based routing like epidemic. Since the size of the *FEs* is quite small (12×4 bytes $= 48$ bytes), the overhead introduced by epidemic routing does not degrade the network performance significantly. Moreover, FN_k can club together multiple *FEs* destined to FN_i, which further reduces the overhead. On receiving $FE_{i,j}(t)$, all other nodes except FN_i keep on forwarding it until the message expires or the evidence reaches FN_i. FN_i on receiving $FE_{i,j}(t)$ in multiple hops, stops forwarding it further. If FN_k gets an opportunity to meet FN_i, it delivers the evidence to FN_i and stops further retransmission. Unlike the end-to-end acknowledgements used in MANETs, *FEs* do not rely on contemporary end-to-end routing paths between the source and the destination and gets transmitted to FN_i in

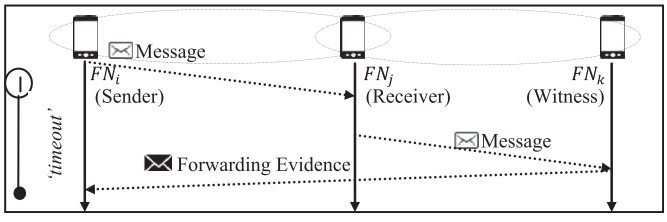

Fig. 2. Forwarding evidence mechanism.

怎么感觉 有点像Eric的two-hop ack

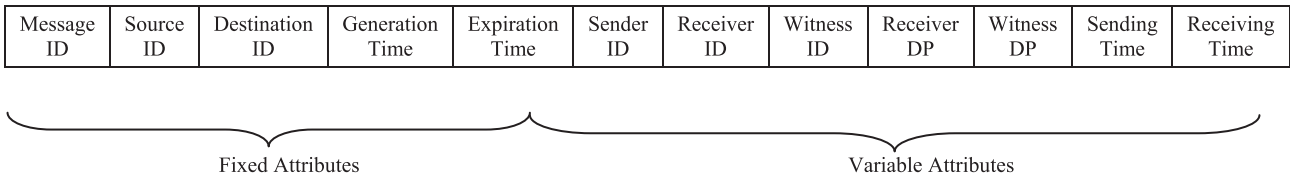

Fig. 3. Forwarding evidence packet structure.

single or multiple hops.

4.1.1.3. FE quantification. On receiving $FE_{i,j}(t)$, all other nodes except FN_i forward it, while FN_i stops forwarding it further. Other nodes also stop forwarding the evidence when the message expires. FN_i verifies the goodness of the forwarding behaviour of FN_j in terms Forwarding Score, $FS_{i,j}(t)$, computed as follows:

$$FS_{i,j}(t) = \begin{cases} 1 & if\ WitnessDP \geq ReceiverDP \\ 0 & otherwise \end{cases}$$

转发分数 forwarding score: 转发给更好的节点，所以给出了 full credit

Therefore, FN_j is given full credit only if it forwards the message to a more competent node. Thus, Forwarding Score is assigned after examining the quality of forwarding. It is to be noted that - (i) FN_j forwards the message received from FN_i to multiple FN_ks using PRoPHET and (ii) each FN_k propagates $FE_{i,j}(t)$ through multiple forwarder-nodes using Epidemic. Owing to the above facts, FN_i may receive multiple $FE_{i,j}(t)$ pertaining to a single message forwarded to FN_j. However, FN_i considers only the $FE_{i,j}(t)$ pertaining to a message that reaches it first. FN_i also increments a Selfishness Score, $SS_{i,j}(t)$, for FN_j as follows:

$$SS_{i,j}(t) = \begin{cases} incremented\ \ if\ FN_i\ does\ not\ receive\ any\ FE_{i,j}(t) \\ within\ 'timeout' \\ \\ not\ incremented\ otherwise \end{cases}$$

4.1.2. Computing direct trust

Forwarding Scores (FS) and Selfishness Score (SS) provided by the Watchdog at the monitoring phase are used to assign direct trust value to the monitored node. For instance, FN_i assigns direct a trust value $T_{i,j}^{direct}(t)$ to FN_j based on all received FSs and SSs about FN_j based on all messages forwarded to FN_j at time t. Such direct trust about FN_j assigned by.

FN_i is computed as:

$$T_{i,j}^{direct}(t) = \begin{cases} 0\ if \sum_{All\ Msg} FS_{i,j}(t) = 0 \\ 0\ \ if \sum_{All\ Msg} SS(t) > \sum_{All\ Msg} FS_{i,j}(t) \\ 1 - \frac{\sum_{All\ Msg} SS_{i,j}(t)}{\sum_{All\ Msg} FS_{i,j}(t)}\ \ otherwise \end{cases} \quad (1a)$$

Each node in the network computes direct trust about all other nodes in the network using equation (1a) and stores them in the Trust Table, shown in Table 1. Direct trust value of nodes that haven't been monitored is set to 0.5.

trust值 初始化为0.5

Updating Direct Trust

FN_i updates direct trust about FN_j according to equation (1b) shown below:

$$T_{i,j}^{'\ direct}(t+\Delta t) = \begin{cases} \mu\ T_{i,j}^{direct}(t) + (1-\mu)\ T_{i,j}^{direct}(t+\Delta t) \\ \qquad if\ FN_i\ can\ monitor\ FN_j in\ \Delta t \\ \\ \Delta e^{-\theta \Delta t}\ T_{i,j}^{direct}(t)\ \ otherwise \end{cases} \quad (1b)$$

时间更新平滑机制

时间衰退

Here, $T_{i,j}^{direct}(t+\Delta t)$ is the direct trust about FN_j at time point $(t+\Delta t)$, assigned by the Watchdog. $T_{i,j}^{'\ direct}(t+\Delta t)$ is the updated direct trust about FN_j at time point $(t+\Delta t)$. Since forwarding behaviour of nodes keep on changing with time between selfish and altruistic, a weighing factor μ, $0 \leq \mu \leq 1$, is used to balance previous and current trust enumerations.

Considering the fact that trustworthiness degrades with time, we multiply direct trust value with a decaying factor $e^{-\theta \Delta t}$, with $0 < \theta \leq 0.1$, if FN_i doesn't get a chance to meet FN_j. These updated direct trust values are disseminated in the network so that nodes can use them in the Indirect Trust Enumeration phase.

Table 1
Trust table stored at FN_i

Node ID		FN_1	…	FN_j	…	FN_N
Direct Trust		$T_{i,1}^{direct}(t)$	…	$T_{i,j}^{direct}(t)$	…	$T_{i,N}^{direct}(t)$
Indirect Trust	Local Trust	$T_{i,1}^{local}(t)$	…	$T_{i,j}^{local}(t)$	…	$T_{i,N}^{local}(t)$
	Global Trust	$\widehat{T}_{i,1}^{global}$	…	$\widehat{T}_{i,j}^{global}$	…	$\widehat{T}_{i,N}^{globaal}$
	Forwarder Trust	$T_{i,1}^{forwarder}$	…	$T_{i,j}^{forwarder}$	…	$T_{i,N}^{forwarder}$
	Rater Trust	$T_{i,1}^{rater}$	…	$T_{i,j}^{rater}$	…	$T_{i,N}^{rater}$

采用这样的形式 来描述？

4.2. Indirect Trust Enumeration

This module consists of four steps. In the first step, a node collects direct trusts about other nodes in the network from its neighbours and uses these direct trusts to compute local trusts. In the second step, the node, using the local trusts, estimates global trusts of all other nodes in the network. As the third step, the node, based on these global trusts, computes forwarder trusts for identifying selfish nodes. In the last step, the collected direct trusts are further used to calculate rater trusts for identifying malicious nodes.

4.2.1. Computing local trusts

When FN_i comes in contact with a group of nodes at particular time, it collects direct trust values (about all other nodes in the network) from these nodes. Suppose, FN_i meets n nodes, say FN_1, FN_2, …, FN_n, at contact time t. FN_i collects direct trust values, $(T_{1,j}^{direct}, T_{2,j}^{direct}, \ldots, T_{n,j}^{direct})$, about FN_j, $j = 1, 2, \ldots, N$, $j \neq i$, from these nodes and uses them to compute a local trust value $T_{i,j}^{local}(t)$ of FN_j as:

$$T_{i,j}^{local}(t) = \frac{1}{n} \sum_{k=1}^{n} T_{k,j}^{direct} \quad (2a)$$

把获得的所有直接结果 求均值
*所有,包括：我自己i 和 被评价的节点j

FN_i stores these local trust values for all other nodes, FN_j, in the network in the Trust Table, shown in Table 1. Local trust values of nodes for which there is no recommendation yet, is set to 0.5.

4.2.2. Estimating global trusts

The direct trust values about a specific node received from various sources may vary to a great extent and there is hardly any global consensus. Therefore, it is logical to consider such direct trust values from all nodes in the network to get the global perception about the specific

确实如此，direct trust 会有很大差别；
试图从direct 获得 全局感知

node. In particular, we define the global trust of FN_j maintained at FN_i as the average of direct trust values about FN_j provided by all $(N-2)$ nodes in the network. Global trust can be represented as:

$$T_{i,j}^{global} = \frac{1}{N-2} \sum_{\substack{k=1 \\ k \neq i,\ j}}^{N} T_{k,j}^{direct} \tag{2b}$$

直接评价求均值
不包括 我自己i 和 被评价节点j

The global trust, in principle, helps a node build a more accurate picture of other nodes' forwarding behaviour. However, in a DTN environment, a node comes in contact with a limited number of nodes at a given time and hence collects a few sample direct trust values about a specific node. Therefore, it is not practicable to collect the entire population of direct trust values and enumerate the global trust based on those. We use statistical estimation (Lehmann and Casella, 1998) to estimate the global trust of a node from a set of sample direct trust values for that node.

实际上 只能收集到一部分、所以采用统计估计的方法

The value of the direct trust (lying between 0 and 1) of a forwarder-node is determined by the events – (i) the forwarder-node transmits a message received from a previous-hop to the next-hop and (ii) a forwarding evidence from the next-hop being received by the previous-hop. Both these events are highly stochastic in nature. Therefore, direct trust can be considered as a continuous random variable and we assume that it follows a uniform distribution over the interval [0, 1] with mean 0.5. The probability density function of direct trust is given by the formula

$$f = \begin{cases} \frac{1}{1-0} = 1, & 0 \leq direct\ trust \leq 1 \\ 0 & otherwise \end{cases}$$

假定 遵守均匀分布？？？？

个人认为 不合适！

For estimating global trust, from a set of sample direct trusts, using statistical estimation, we consider the following:

Total number of nodes: N
Population size: N − 2, excluding FN_i and. FN_j
Population: $\{T_{1,j}^{direct}\}$, $i = 1, 2, \ldots, N-2$
Population mean: $T_{i,j}^{global}$, global trust of FN_j to be stored at. FN_i
Sample size: n
Sampling scheme: Simple random sampling without replacement.
Probability of a sample: $1/{}^{N}C_n$
Sample: $\{T_{i,j}^{direct}\}$, $i = 1, 2, \ldots, n$,
Sample mean: $T_{i,j}^{local}$, to be stored at. FN_i
Following the theory of random sampling.

无放回 抽样

可以用在我们现在的NN-Detect paper里

也可以用在Trust-Min paper里

$T_{i,j}^{local}$ can be considered as a random variable that assumes different values for different samples with probability $1/{}^{N}C_n$. In the proposed technique, FN_i repeats the simple random sampling S times at consecutive time points $(t, t+\Delta t, \ldots, t+(S-1)\Delta t)$ and computes local trust values of FN_j as $\{T_{i,j}^{local}(t), T_{i,j}^{local}(t+\Delta t), \ldots, T_{i,j}^{local}(t+(S-1)\Delta t)\}$. These local trusts, i.e., the different values of the sample mean $T_{i,j}^{local}$, each occurring with probability $1/{}^{N}C_n$, are stored in a Sampling Distribution of Local Trust, shown in Table 2.

Now according to the theory of statistical estimation, the sample mean is an unbiased estimator of the population mean and expectation of sample mean is an unbiased estimate of the population mean. In our context, local trust, $T_{i,j}^{local}$, is an unbiased estimator of global trust, $T_{i,j}^{global}$

Table 2
Sampling distribution of local trust values of FN_j

$T_{i,j}^{local}$	$T_{i,j}^{local}(t)$	…	$T_{i,j}^{local}(t+(S-1)\Delta t)$
Probability	$1/{}^{N}C_n$	…	$1/{}^{N}C_n$

实际上direct值 会有随时间变化的趋势//
这个趋势 本身的稳态 才能真正代表 结果(现实环境最终的样子)！！！！

and

只考虑到了 估计 确没有考虑到趋势。。。
从不清楚逐渐变得清楚

$$\widehat{T}_{i,j}^{global} = E\left(T_{i,j}^{local}\right) = T_{i,j}^{local}(t) \times 1/{}^{N}C_n + \cdots + T_{i,j}^{local}(t+(S-1)\Delta t) \times 1/NC_n \tag{2c}$$

The accuracy of the estimated global trust depends upon the number of samples (S, in our case), i.e., the number of local trusts, being considered for estimation. FN_i similarly estimates the global trust values of all other nodes in the network and stores them in the network in the Trust Table, shown in Table 1. Such estimated global trust is then combined with the direct trust to obtain a comprehensive forwarder trust value for a node.

4.2.3. Combining direct and global trusts to form forwarder trust

To judge the forwarding behaviour of FN_j, FN_i combines the direct and global trusts FN_j to form a forwarder trust value as

$$T_{i,j}^{forwarder} = \omega_1 T_{i,j}^{direct} + \omega_2 \widehat{T}_{i,j}^{global} \tag{2d}$$

where $T_{i,j}^{direct}$ and $T_{i,j}^{global}$ are the last updated direct trust and global trust of FN_j maintained by FN_i. The weighing factors ω_1 and ω_2 $(0 \leq \omega_1, \omega_2 \leq 1,\ \omega_1 + \omega_2 = 1)$ are used to decide the importance of direct observation against recommendations. FN_i stores these direct trust values for all other nodes in the network in the Trust Table, shown in Table 1. These forwarder trust values are used for enumerating the forwarding competency of a prospective forwarder which is explained in Section 4.3.

4.2.4. Computing rater trust

针对collusion

To mitigate bad-mouthing and ballot-stuffing, FN_i judges the accuracy of the direct trust values provided by FN_j by computing its rater trust. To compute rater trust, FN_i calculates the deviations of direct trust values shared by FN_j, about other nodes, from the average direct trust value (i.e., local trust value) of these nodes. These deviations indicate to what extent recommendations provided by FN_j diverge from the recommendations provided by others. FN_i finds the mean of deviations (MD) at time t as below:

$$MD_{i,\ j}^{direct}(t) = \frac{1}{N-2} \sum_{\substack{l=1 \\ l \neq i,j}}^{N} \left| T_{j,l}^{direct}(t) - T_{i,l}^{local}(t) \right| \tag{2e}$$

j自己给自己提供的偏差

To bring rationality to the judgment process, FN_i computes the mean of deviations for S consecutive time points. The rater trust value is then computed as:

$$T_{i,j}^{rater} = 1 - \frac{1}{S} \sum_{m=0}^{S-1} MD_{i,\ k}^{direct}(t+m\Delta t) \tag{2f}$$

从时间的维度上看待这种偏差

If value of $T_{i,j}^{rater}$ falls below a specified threshold τ (set through simulation), FN_i considers FN_j as malicious and refrains from using the direct trust values it provides. FN_i stores rater trust values for all other nodes in the Trust Table, shown in Table 1.

从时间上看 时间上累计的偏差大于阈值 就会被认为是谎言。。。

4.3. Forwarder Selection

In this module, a node first enumerates the forwarding competency of a prospective forwarder and uses this competency to select the best next-hop forwarder. Forwarder trust of a node determines its forwarding behaviour (selfish or altruistic). Its delivery predictability, on the other hand, depicts the probability of meeting the final destination of the message. Therefore, FN_i while taking forwarding decision integrates these metrics to find Forwarding Competency ($FC_{i,j}$) of the forwarder FN_jas shown below:

$$FC_{i,\ j} = T_{i,j}^{forwarder} \times P(FN_j, DestID) \quad (3)$$

Whenever FN_i meets FN_j and intends to forward a message, FN_i computes $FC_{i,j}$ and $FC_{i,i}$. FN_i takes forwarding decision according to the following rule:

Forwards message to FN_j if. $FC_{i,j} > FC_{i,i}$
Don't forward message to FN_j, otherwise.

The rationale behind this multiplicative model in equation (3) is that if FN_j is selfish, FN_i assigns a low forwarder trust to FN_j and $FC_{i,\ j}$ is decreased. However, this model suffers from *self-trusting*, a property of trust based data forwarding in opportunistic networks. Self-trusting refers to the fact that nodes trust themselves more than others (Li and Das, 2013), implying that FN_i sets $T_{i,i}^{forwarder}$ to 1 in all situations. Therefore, the forwarding competency of FN_i to itself, i.e., $FC_{i,\ i}$, is only equal to its delivery predictability $P(FN_i, DestID)$. As a result, FN_i attempts to directly deliver messages to the destination itself, unless it encounters another node with much stronger delivery predictability and/ or extremely high forwarder trust. The effect of self-trusting is increased delay in delivering messages to the destination, as most of the nodes get rejected as competent forwarders. This is demonstrated in the quantitative performance analysis section.

To combat the effect of self-trusting, a self-trust rationalizing factor, say α is introduced. Whenever FN_i meets FN_j and intends to forward a message, FN_i computes $FC_{i,j}$ using equation (3) and computes $FC_{i,i}$ as

$$FC_{i,\ i} = (\alpha \times T_{i,i}^{forwarder}) \times P(FN_i, DestID) \quad (4)$$

where α ranges from 0.5 to 1. Assuming α as 0.5 will allow FN_i to select forwarders with moderate delivery predictability and forwarder trust. This results in faster delivery of more number of messages, although at the risk of incorporating some selfish nodes into the forwarding activity. Taking α as 1 compels FN_i to choose forwarders with high delivery predictability and/or high forwarder trust. This results in slower delivery of lesser number messages, ensuring the elimination of selfish nodes from the forwarding activity.

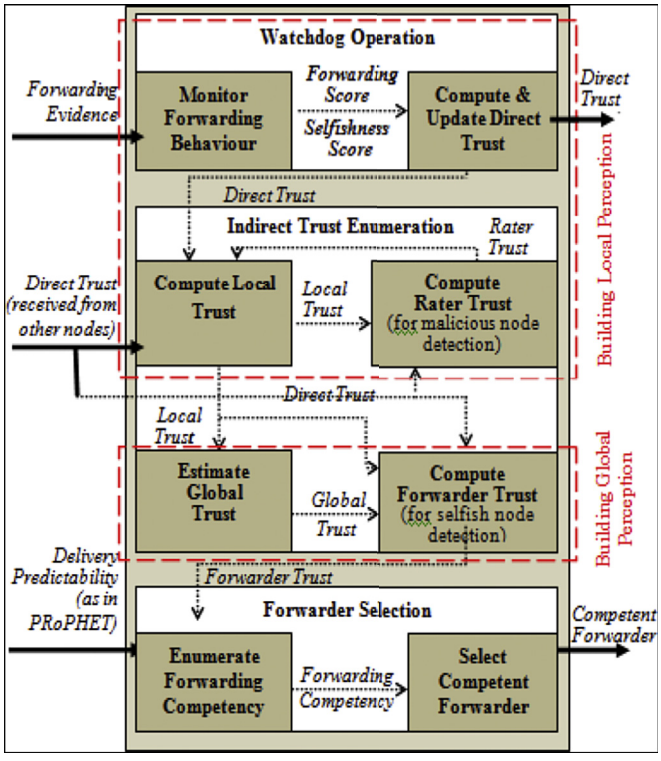

Fig. 4. The proposed Wise-PRoPHET framework.

the forwarding activity.

The three interoperable modules of Wise-PRoPHET work together for identifying selfish and malicious nodes in the network and forwarding post disaster situational messages through the best possible forwarders. Fig. 4 elucidates the operation technique.

4.4. *The Wise-PRoPHET algorithm*

In this section, we present a global view of the algorithm for the Wise-PRoPHET technique discussed above.

Input:	Forwarding Evidence, Direct Trust, Delivery Predictability
Output:	Direct Trust, Competent Forwarder
Runs At:	Forwarder-node (FN_i, DTN enabled smartphone, carried by volunteers and relief workers)

// **WATCHDOG OPERATION BEGINS**

if *Rater Trust of* $FN_j > \tau$ then
  accept direct trust values from FN_j
  endif
  share direct trust values with FN_j
for $MID = 1$ *to* M **do** // M Messages
// Direct Trust Enumeration
if *FE for MID received within 'timeout'* **then**
  if *WitnessDP* $\geq$ *ReceiverDP* **then**
  Forwarding Score $= 1$
  else
  Forwarding Score $= 0$
  endif
  else
  Increment Selfishness Score
  endif
  end for
  Compute Direct Trust of FN_j //*using* Eq (1a)
  Update Direct Trust of FN_j //*using* Eq (1b)

// **WATCHDOG OPERATION ENDS**
// **INDIRECT TRUST ENUMERATION BEGINS**
// Local Trust Computation
Compute Local Trust of FN_j //*using* Eq (2a)
// Global Trust Estimation
Estimate Global Trust of FN_j //*using* Eq (2c)
// Forwarder Trust Computation
Compute Forwarder Trustof FN_j //*using* Eq 2(d)
// Rater Trust Computation
for $m = 0$ to $S - 1$ **do**// assuming S timepoints
  Compute Mean of Deviations //*using* Eq (2e)
  endfor
  Compute Rater Trust of FN_j //*using* Eq (2f)
end for

// **INDIRECT TRUST ENUMERATION ENDS**
// **FORWARDER SELECTION BEGINS**
for *MID* $= 1$ *to* M **do** // M Messages:
  Compute Forwarding Competency of FN_j *using* Eq (3)
  if $FC_j > FC_i$ **then**
  transfer *MID* to FN_j
  endif
  end for

// **FORWARDER SELECTION ENDS**

4.4.1. *Algorithmic complexity*

In the worst case, the Watchdog Operation module runs for a maximum of M times for all messages (M). So, the complexity is $O(M)$. In the Indirect Trust Enumeration module, local trust computation runs S times corresponding to the S time points. Similarly, rater trust computation also runs for S times. Thus, the complexity can be derived as $O(S + S) = O(2S) = O(S)$.

Each of Watchdog Operation, Indirect Trust Enumeration, and Forwarder Selection modules run for a maximum $(N - 1)$ times, as a forwarder-node FN_i can meet at most $(N - 1)$ other nodes in the network. Therefore, the worst-case complexity of the algorithm is

$$O(N(M + S))$$

4.4.2. *Correctness of the algorithm*

The trust based forwarding competency of a node is used to prove the

markov 吸收态！！

correctness of the Wise-PRoPHET algorithm.

Lemma. The forwarding competency of a next–hop forwarder determines whether a message gets forwarded to it.

Proof. Let A be the event that the next-hop forwarder of a node is not selfish. Then A^c is the event that it is selfish. Also, let B be the event that the message is forwarded to the next-hop forwarder. Since, A and A^c are mutually exclusive and exhaustive, we have,

$$B = (A \cap B) \cup (A^c \cap B)$$

Since the events $(A \cap B)$ and $(A^c \cap B)$ are also mutually exclusive, applying the theorem of total probability, we have

$$P(B) = P(A \cap B) + P(A^c \cap B)$$

Applying theorem of compound probability, we get

本文把prob当成了一个随机的值

$$P(B) = \{P(A) \times P(B/A)\} + \{P(A^c)) \times P(B/A^c)\} \quad (5)$$

The probability that the message gets forwarded to the next-hop forwarder, given that it is not selfish, is determined by the delivery predictability of the forwarder$P(next - hop, DestID)$, as described in Section 3.3. Therefore,

$$P(B/A) = P(next - hop, DestID).$$

Also, probability that the message gets forwarded to the next-hop forwarder, given that it is selfish is 0,i.e., $P(B/A^c) = 0$

Also, $P(A) = T^{forwarder}_{next-hop}$ and $P(A^c) = 1 - T^{forwarder}_{next-hop}$.

Putting these values in equation (5), we get

$$P(B) = T^{forwarder}_{next-hop} \times P(next - hop, DestID)$$

which is, in fact, the forwarding competency of the next–hop forwarder derived in equation (3). Therefore, the trust based forwarding competency of a next-hop forwarder determines whether a message gets forwarded to it. This proves the correctness of our technique.

等待ack会有一个超时时间

4.4.3. Deriving a rational 'timeout' period

As explained in Section 4.1.1.1, FN_i after sending a message to FN_j at time t, waits for a *timeout* period for the arrival of a forwarding evidence $FE_{i,j}(t)$ pertaining to the message. Return (or non-return) of the forwarding evidence within the *timeout* period has a major impact on the direct trust about FN_j, as explained in equation (1a). Therefore, it becomes imperative to derive a rational *timeout* period for which FN_i must wait before incrementing the selfishness score of FN_j.

从epidemic角度得出延迟就是timeout时间的最好估计

As illustrated in Fig. 2, *timeout* should be the total time required for the message to reach FN_k from FN_j (using the Wise-PRoPHET technique) and the forwarding evidence to reach FN_i from FN_k (using epidemic routing). So, we attempt to compute the expected message delay first for epidemic routing and then for the proposed Wise-PRoPHET technique. Transmissions between two nodes are assumed to take place at meeting times of the nodes and instantaneous, i.e., the transmission time of a message is very small with respect to the meeting times. Hence, message delay refers to the time required for a node to find the next possible forwarder and deliver the message to that forwarder. To compute the expected message delays, we introduce a stochastic model that models message delay between any two nodes in a DTN. We adapt the works presented in (Groenevelt et al., 2005) and (Haas and Small, 2006) to derive expressions for expected message delay in our proposed technique.

We consider the network with N identical forwarder-nodes, as stated in Section 3.2. Suppose there is a single message to be delivered by a source node to a destination node. Intermediary nodes are used as relay nodes. Let $0 \le t_{p,q}(1) < t_{p,q}(2) < \ldots$ be the successive meeting times between forwarder-nodes FN_p and FN_q. We define, $\delta_{p,q} = t_{p,q}(n+1) - t_{p,q}(n)$, as the n^{th} inter-contact time between FN_p and FN_q. The processes, $\{t_{p,q}(n), n \ge 1\}, 1 \le p, q \le N, p \ne q$, are mutually independent and identically distributed Poisson processes with rate $\lambda > 0$ (Groenevelt et al., 2005). Equivalently, the random variables $\{\delta_{p,q}\}$ are mutually independent and exponentially distributed with mean $1/\lambda$, $\lambda > 0$ is the parameter of the distribution, often called the rate parameter and $1/\lambda$ average inter-contact length. The number of copies of the message in the network can be modeled as an absorbing finite-state Markov chain. The Markov chain takes its value in $\{1, 2, \ldots, N\}$. The Markov chain is in state $i = 1, 2, \ldots, (N-1)$ when there are i copies of the message in the network including the original message, and it is in state N when the message is delivered to the destination node. The states $1, 2, \ldots, (N-1)$ are transient states whereas N is an absorbing state.

The transition diagram of the Markov chain for a multicopy routing protocol (controlled or uncontrolled) is shown in Fig. 5. In this protocol, each node which has a copy of the message forwards it to a node that does not have a copy and which comes within its transmission range. The message is forwarded to an encountered node with a probability P if it is not the destination and with a probability 1 if it is the destination. Therefore, when there are i copies of the message in the network, a new copy is created at the rate $iP\lambda(N-1-i)$ (transition from state i to $i+1$) and one of those i copies reaches the destination node at the rate $i\lambda$ (transition from state i to N), as illustrated in Fig. 5. The chain jumps from state i to $i+1$ with probability $P(N-i-1)/(P(N-i-1)+1)$ and it jumps from state i to N with probability $1/(P(N-i-1)+1)$.

In Wise-PRoPHET (controlled multicopy routing), a node forwards the message to its next-hop forwarder with a probability P, where

$$P = T^{forwarder}_{next-hop} \times P(next - hop, DestID)$$

if the next-hop forwarder is not the destination and with a probability 1 if it is the destination. Using results from (Groenevelt et al., 2005), we derive the expected message delay (*EMD*) for Wise-PRoPHET as:

$$EMD_{Wise-PRoPHET} = \frac{1}{\lambda(N-1)} \sum_{i=1}^{N-1} \sum_{j=1}^{i} \frac{1}{PNj - Pj^2 + Pj + j} \quad 6(a)$$

which time taken by the message to reach FN_k from FN_j. P, lying between 0 and 1, differs for each next-hop forwarder.

In epidemic (uncontrolled multicopy) routing, a node forwards the message to its next-hop forwarder with a probability 1 irrespective of whether it is the destination or not, i.e., for such a routing $P = 1$. Using results from (Groenevelt et al., 2005), we derive the expected message delay (*EMD*) for epidemic routing as:

$$EMD_{epidemic} = \frac{1}{\lambda(N-1)} \sum_{i=1}^{N-1} \frac{1}{i} \quad 6(b)$$

which is the time taken by the forwarding evidence to reach FN_i from FN_k. Therefore, a rational value for *timeout* can be obtained by combining the results derived in equations (5a) and (5b) as

$$timeout = EMD_{Wise-PRoPHET} + EMD_{epidemic} \quad 6(c)$$

It can be observed from equation (6a) that the minimum value of $EMD_{Wise-PRoPHET}$ is $\frac{1}{\lambda(N-1)} \sum_{i=1}^{N-1} \frac{1}{i}$, obtained when $P = 1$. Thus, using equation (6c), we obtain the minimum value of *timeout* (i.e., the minimum time for which FN_i must wait before considering FN_j as selfish) as $\frac{2}{\lambda(N-1)} \sum_{i=1}^{N-1} \frac{1}{i}$. For practical values of $1/\lambda = 1800$ secs, 3600 secs, 7200 secs (Guo and Chan, 2013) and $N = 100$, the minimum value of *timeout* comes out to be 180 secs, 360 secs and 720 secs respectively. We use these *timeout* values for quantitative performance evaluation of the proposed technique in Section 5.2.

5. Performance evaluation

The effectiveness of the proposed Wise-PRoPHET technique reported

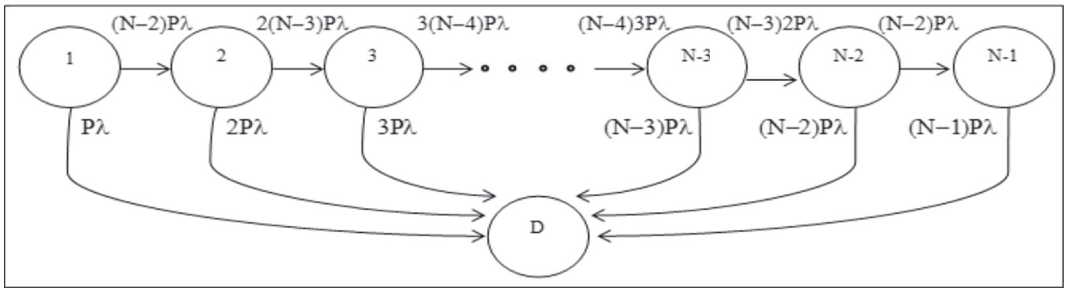

Fig. 5. Transition diagram of the Markov chain for multicopy routing.

in the earlier section is evaluated by both qualitative and quantitative analyses.

5.1. Qualitative analysis

This section provides overhead analysis of the Wise-PRoPHET technique which is run on resource constrained devices like smartphones (forwarder-node). We measure the worst case computation and communication overheads for an operational period, which is typically 12 h for our application. The worst case scenario arises when the node meets all other $(N-1)$ nodes in the network. We assume that a node on an average has P contacts with every other node, meets n nodes at time t and exchanges M messages. Storage overhead is not evaluated considering the substantial storage facilities in smartphones. In our analysis, we consider the Samsung Galaxy Grand 2 smartphone specification. As per specification, the smartphone has a Qualcomm *Snapdragon* processor with ARM core. Battery capacity is 2600 mAh with 3.8 V and initial energy of 35568 J.

5.1.1. Computation overhead

The computation overhead of a forwarder-node for executing Wise-PRoPHET is the total number of low-end operations (Wang and Zwolinski, 2013) it performs. Table 3 shows the required number of such low-end operations and corresponding energy requirements. Using these, the computation energy overhead sums up to:

$$3.5MN + 0.9Mn + 2.3NS + 0.2n^2 - 4.4M + 1.5N - 2.1S + 0.2n - 1.8 \text{ Joules}$$

5.1.2. Communication overhead

We evaluate the communication overhead in terms of the total number of bytes exchanged by a forwarder-node for executing the Wise-PRoPHET technique. As per the assumption stated above, a node can meet all other $(N-1)$ nodes in the network with P contacts per node. On each such contact the forwarder-node exchanges direct trust values of $(N-1)$ other nodes. Therefore the node transmits and also receives $((N-1)(N-2)P)$ direct trust values. Assuming M transmitted and Q received Forwarding Evidences in the entire operational period, a forwarder-node transmits $12M$ and receives $12Q$ values in the entire operational period. Therefore, assuming k bytes for each transmitted/received value, communication overhead (in bytes) sums up to:

$$\text{Transmission Overhead: } k\left(N^2P - 3NP + 12M + 2P\right)$$

$$\text{Reception Overhead: } k\left(N^2P - 3NP + 12Q + 2P\right)$$

The Bluetooth communication energy consumption for the smartphone under consideration is 0.384 J/ sec for transmitting and 0.329 J/ sec for receiving 1 Mb data (Bhattacharjee et al., 2014). Also, Bluetooth 1.2 has an average data transfer rate of 1 Mb/s. Thus, the communication energy overhead is computed as.

Transmission Energy Overhead:

$$0.384/(1024)^{-2} \times k\left(N^2P - 3NP + 12M + 2P\right) \text{ Joules}$$

Reception Energy Overhead:

$$0.329/(1024)^{-2} \times k\left(N^2P - 3NP + 12Q + 2P\right) \text{ Joules}$$

Illustrative Example

Assuming $k=4,\ M=50,\ N=50,\ P=10,\ S=5,\ Q=15, n=5$ the energy consumption for running the proposed technique for the entire operational period sums upto 2736.15 *Joules*. The energy consumed for running background processes is 0.562 *Joules*/sec and initial energy of the smartphone is 35568 *Joules*. Hence, a smartphone dedicated for running the proposed technique can function for almost 13 h for our application, i.e., for the full operational period in a day, without further charging.

Table 3
Operations and corresponding energy consumption.

Operation	Number of Operations	Energy Consumption (Joules)/Operation
LOAD	$7MN + 6NS - 7M + 2N - 6S - 2$	0.23
MOV	$4MN + NS - 4M + 4N - S - 4$	0.20
STORE	$3MN + -3M + 2N - 2$	0.12
CMP	$MN + NS - M - S$	0.75
MUL	$5Mn + n^2 - 5M + n + S - 2$	0.18

5.2. Quantitative analysis

The effectiveness of the proposed Wise-PRoPHET technique is evaluated through simulation.

5.2.1. Simulation environment

We use a real disaster scenario for setting up the simulation environment, based on the 2015 Nepal earthquake. The Google Map of water, food, shelter and medical resources for Nepal earthquake (Map of water, food, shelter, etc.), shown in Fig. 6, marks the shelters and medical relief centers set in Kathmandu and its adjoining districts like Nuwakot, Sindhupalchowk, and others.

We set up our simulation environment based on the above snapshot.

5.2.2. Simulation setup

We create our simulation set-up in ONE simulator (Keranen et al., 2009), based on information provided by the map in (Map of water et al., 2015) regarding post-disaster relief operation carried out in Nepal, at the Kathmandu area. 9 shelters are set up on the map of the disaster affected area of Kathmandu, an area of 8.5 square km. The shelters are set up in Durbar Square, St. Xavier's School, Pulchowk Engineering Campus, Nepal Academy of Science, National Agriculture Research Centre, Jawalakhel Football ground, etc. The control station is setup at Tribhuvan University.

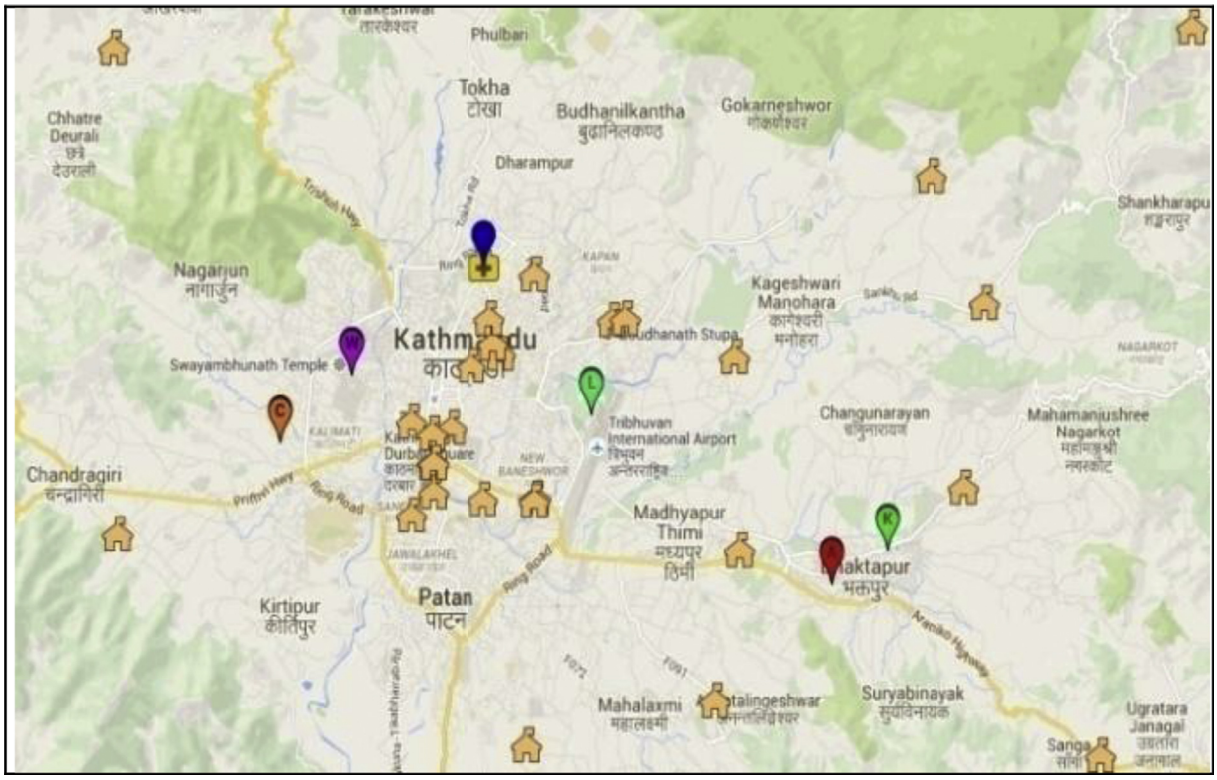

Fig. 6. Google map for the 2015 Nepal earthquake (Map of water, food, shelter, etc.).

We assume 10 forwarder-nodes, i.e., volunteers with smartphones working in each shelter who move across the entire affected area, exchange situational messages of the shelters and transmit them to the control station. The shelter-nodes and the control-node are stationary. However, the forwarder-nodes are mobile nodes and follow the Post Disaster Mobility (PDM) Model proposed by Uddin et al. in (Uddin et al., 2009 Doctors For You. http://d). Post disaster relief operations are usually carried out during the daytime, typically from 6 a.m. to 6 p.m. Accordingly, we keep our simulation time 12 h. A shelter-node generates situational messages at a rate of one per hour and broadcasts them towards the forwarder-nodes. We use real situational messages derived from the WhatsApp chat-log of volunteers of an NGO, "Doctors For You" (Doctors For You. http://d Doctors For You. http://d), who provided medical relief during and after the earthquake. These messages, containing the actual requirements of emergency resources at different shelters, are 120 bytes in size. The Forwarding Evidences (Section 4.1.1.1) are of size 48 bytes. We consider benign, selfish (*SN*) and malicious (*MN*) nodes in our network. Benign nodes participate in routing activity following the proposed Wise-PRoPHET technique. Selfish nodes drop all messages (destined to other nodes) forwarded to it. Malicious nodes set the direct trust value to 1 for all nodes whose direct trust is less than 0.5 using equation (1a) and set direct trust value to 0 for all nodes whose direct trust is more than or equal to 0.5. We set the *timeout* period (time for which a node must wait before identifying its neighbour as selfish) as 360 secs for $1/\lambda = 3600$ secs. The basis of setting such value for *timeout* is as per the guidelines provided in Section 4.4.3. Table 4 lists the important parameters used in simulation.

5.2.3. Simulation metrics

Performance of Wise-PRoPHET is evaluated in two stages: measuring the extent of achieving major design goals and evaluation of network performance.

Metrics used for measuring the extent of achieving major design goals are:

Estimation Bias – Bias of an estimator is the difference between the estimator's (local trust, in our case) expected value and the true value of the parameter (global trust, in our case) being estimated. Thus, estimation bias indicates how accurately the proposed technique can estimate the global trust of a forwarder and is defined as $Estimation\ Bias = |E(local\ trust) - global\ trust|$ where $E(local\ trust)$ denotes the expectation of local trust, defined in equation (2c). *Detection Ratio* – Indicates how well the proposed technique can detect selfish nodes (Chen et al., 2013). We consider two types of such ratios:

$$Ratio\ of\ True\ Positive = \frac{No.\ of\ correct\ detections}{No.\ of\ selfish\ nodes\ in\ the\ network}$$

$$Ratio\ of\ False\ Positive = \frac{No.\ of\ false\ detections}{No.\ of\ nodes\ detected\ as\ selfish}$$

Attraction Ratio – Indicates how well the proposed technique can restrict

Table 4
Parameter used for simulation.

Parameter		Value
Simulator Parameters+	Simulation period	12 h
	No. of shelters	9
	No. of nodes	90
	Node speed	10-50 kmph
	Transmission Range	10 m
	Message Generation Time	One per hour
	Message TTL	500 min
	Buffer size	5 MB
Wise-PRoPHET Parameters	*timeout*	360 s
	μ, ω_1, ω_2	[0, 1]
	θ	[0, 0.1]
	S	[4 to 10]
	τ	0.5
	α	[0.5, 1]

forwarding messages to selfish nodes (Dias et al., 2015). Defined as

$$Attraction\ Ratio = \frac{No.\ of\ msgs.\ received\ by\ selfish\ nodes}{No.\ of\ msgs.\ relayed}$$

Metrics (Wang et al., 2014) used for evaluation of network performance are:Delivery Ratio – It is the fraction of the messages delivered to destination nodes to those created by source nodes. Defined as

$$Delivery\ Ratio = \frac{No.\ of\ msgs.\ delivered}{No.\ of\ msgs.\ created}$$

Average Delay – Indicate the average time taken by the messages from sources to destinations, including buffer delays, queuing delays, retransmission delays and propagation time. Defined as

$$Average\ Delay = \frac{\sum_{for\ all\ M_D}(T_D - T_S)}{No.\ of\ msgs.\ delivered}$$

Overhead Ratio – Indicates the ratio of the number of control messages (including route request/reply/update/error packets) to the number of data messages. Defined as

$$Overhead\ Ratio = \frac{No.of\ control\ packets\ relayed}{No.\ of\ data\ packets\ relayed}$$

where T_D is the time when message reaches its destination and T_S is the time when the message was created at the source.

Finally, we define *Tolerance Level* to evaluate the comprehensive performance of our technique. It is defined as the maximum percentage of selfish and malicious nodes that can be tolerated for achieving a given level of network performance, in terms of delivery ratio, average delay, and overhead ratio.

5.2.4. Results and discussion

We compare our technique with two other competing techniques T-Threshold (Chen et al., 2013) and T-PRoPHET (Li and Das, 2013), in terms of both design goals and network performance. T-PRoPHET does not deal with bad-mouthing and ballot-stuffing and hence we consider the percentage of malicious nodes in T-PRoPHET as 0. The results presented here are the average of 50 independent runs.

5.2.4.1. Achieving major design goals. Four sets of experiments are conducted for evaluating the performance of the present technique by measuring the extent of attaining design objectives.

As mentioned in Section 4.2.2, the accuracy of the estimated global trust depends upon the number of samples (S, in our case) being considered for estimation. Hence, in the first set of experiment, we measure the average estimation bias across all nodes against the number of samples to figure out the optimum number of samples to be considered. We consider a fixed (50%) number of selfish and malicious nodes in the network. As expected, Fig. 7 exhibits that the average Estimation bias reduces with increasing sample size. It is observed that a sample size of 4 reduces the estimation bias to less than 10%. Accordingly, we conduct all further experiments by varying the sample size from 4 to 10.

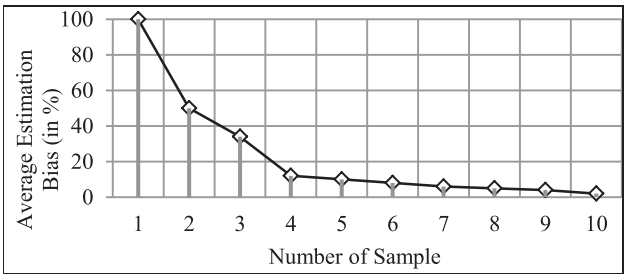

Fig. 7. Bias in estimating global trust against sample size.

The speed, at which a forwarder-node generates the global perception about the forwarding behaviour of its neighbours, has a direct impact on the detection of selfish nodes. Thus, in the second set of experiment, we evaluate the timeliness of global trust generation at each forwarder-node. We use the simulation results on estimation bias, obtained in the first set of experiment, to compute the global trust accuracy at each node. Fig. 8 shows the average global trust accuracy across all nodes against time, with a fixed (50%) number of selfish nodes. It is observed that the average global trust accuracy reaches 90% after 7 h of simulation in presence of 50% malicious nodes, Fig. 8(a); reaches 90% after 6 h of simulation in presence of 25% malicious nodes, Fig. 8(b); and reaches 90% after 4 h of simulation in absence of malicious nodes, Fig. 8(c).

In the third set of experiment, we measure variation in detection ratios. These ratios are directly linked with bad-mouthing and ballot-stuffing, hence varies with the percentage of malicious nodes. Fig. 9 plots detection ratios with a varying number of malicious nodes and a fixed (50%) number of selfish nodes. Detection ratio improves with reducing number of malicious nodes in both the techniques. Improvement rate in our technique is higher compared to Trust-Threshold. Precisely, Fig. 9(a) indicates that Wise-PRoPHET out-performs Trust-Threshold beyond 35% malicious nodes and has an average of 10% more true detections. Fig. 9(b) shows that Wise-PRoPHET performs consistently better and has an average of 12% less false detections than Trust-Threshold.

The fourth set of experiment measures attraction ratio for three competing techniques, assuming 50% malicious nodes. Fig. 10 shows that all three techniques show a reducing trend, but T-PRoPHET performs best owing to the absence of malicious nodes. Fig. 10(a) shows that Wise-PRoPHET has an average of 7% better performance than Trust-Threshold. Fig. 10(b) justifies that Wise-PRoPHET has an average of 9% better performance than Trust-Threshold.

5.2.4.2. Evaluation of network performance. To evaluate network performance while achieving the design goals, 5 sets of experiments are conducted.

As mentioned in Section 4.3, the number of messages delivered and its speed through the Wise-PRoPHET technique depends on the value of the self-trust rationalizing factor (α). Hence, in the first set of experiment, we attempt to figure out the optimum value of α that should be considered for conducting further experiments on Deliver Ratio and Average Delay. Fig. 11(a) and (b) plot Delivery Ratio and Average Delay respectively, of Wise- PRoPHET against different values of α. Average of.

All the results obtained for different percentages of selfish and malicious nodes is plotted. In one hand, it is observed that, for smaller values of α, Delivery Ratio reduces due to the inclusion of selfish nodes resulting in frequent packet drops. On the other hand, for bigger values of α, the ratio reduces due to the effect of self-trusting. Also, Average Delay increases consistently with increasing value of α, as nodes with high delivery predictability and/or high forwarder trust are chosen for forwarding messages. Delivery Ratio is the highest for $\alpha = 0.7$. There-

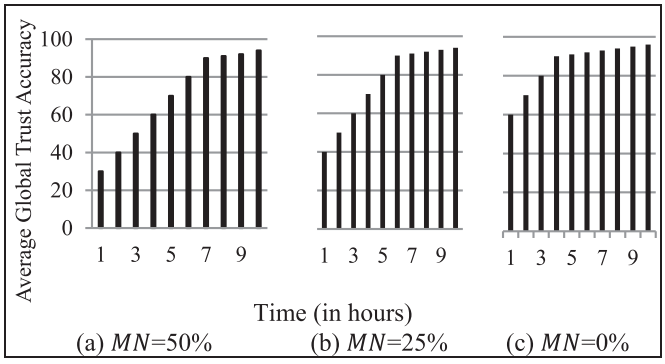

Fig. 8. Average global trust accuracy over time.

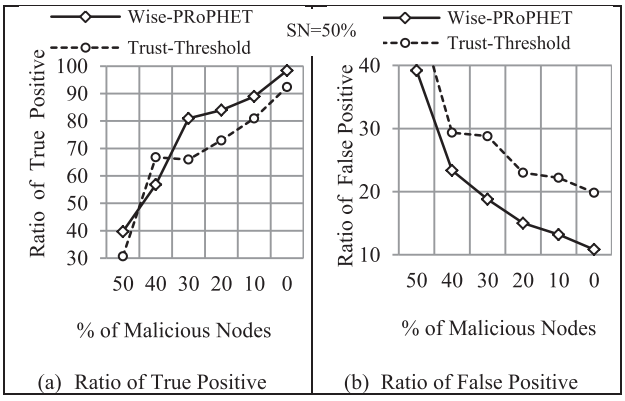

Fig. 9. Detection Ratios with different levels of maliciousness.

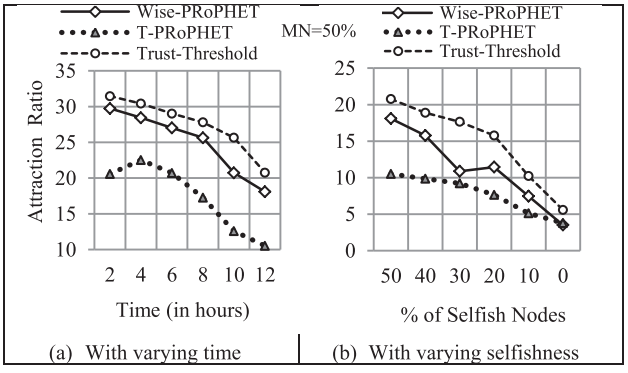

Fig. 10. Attraction ratio.

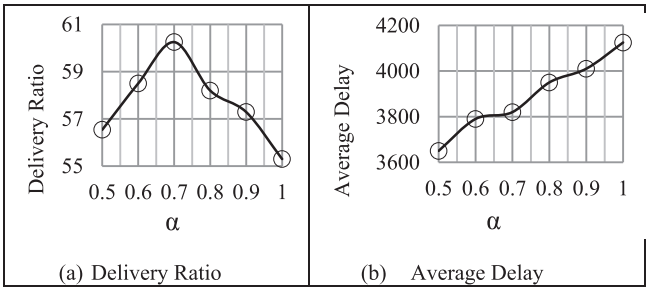

Fig. 11. Impact of self-trust rationalizing factor.

fore, we conduct all further experiments with $\alpha = 0.7$.

In the second set of experiment, the performance of Wise-PRoPHET is compared with other competing schemes in terms of Delivery Ratio. T-PRoPHET performs the best, as the percentage of malicious nodes in T-PRoPHET is 0. However, T-PRoPHET beats Wise-PRoPHET by only 3.25% in presence of 50% malicious nodes, Fig. 12(a). It beats by only 0.18% in presence of 25% malicious nodes, Fig. 12(b). Wise-PRoPHET beats T-PRoPHET by 25.5% and Trust-Threshold by 23% in absence of malicious nodes, Fig. 12(c). Thus, Wise-PRoPHET performs almost as good as T-PRoPHET in presence of malicious nodes and outperforms it in absence of such nodes. This is because T-PRoPHET suffers from the self-trusting effect and Wise-PRoPHET reduces this effect by choosing $\alpha = 0.7$. Moreover, Wise-PRoPHET, unlike others, uses global trust values, for computing forwarding competencies, instead of only direct trust values.

In the third set of experiment, the performance of Wise-PRoPHET is compared with other schemes in terms of Average Delay. Trust-Threshold performs the best. However, the average delay is only 8.8%

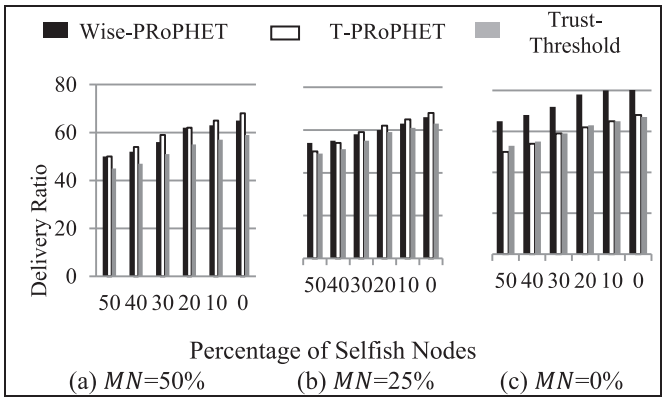

Fig. 12. Delivery Ratio with different levels of selfishness.

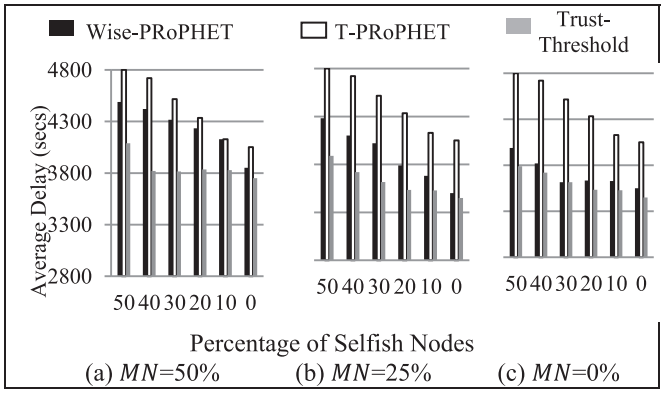

Fig. 13. Average Delay with different levels of selfishness.

less than Wise-PRoPHET in presence of 50% malicious nodes, shown in Fig. 13(a); 6.7% less in presence of 25% malicious nodes, shown in Figs. 13(b) and 2.6% less in absence of malicious nodes, shown in Fig. 13(c). The reason for this excess delay in Wise-PRoPHET is the message delay incurred due to the Forwarding Evidences, but then these evidences make Wise-PRoPHET better than Trust-Threshold in terms of Detection Ratio, Attraction Ratio, and Delivery Ratio. However, Wise-PRoPHET outperforms T-PRoPHET owing to the use of $\alpha = 0.7$.

The fourth set of experiment measures Overhead Ratio of all the three techniques. For Wise-PRoPHET, control packets refer to the Forwarding Evidences, for T-PRoPHET they refer to Positive Feedback Messages and for Trust-Threshold these are the different types of tokens. Fig. 14 shows that for the Wise-PRoPHET technique *FE* packets can be at most 14% of all transmitted packets. Multiple *FE*s destined to a specific node are clubbed together to reduce the overhead. T-PRoPHET outperforms Wise-PRoPHET by 4% in case of 50% malicious nodes.

Fig. 14(a), and by 3% in case of 25%s malicious nodes, Fig. 14(b). However, both Wise-PRoPHET (by 18%) and Trust-Threshold (by 5%) beats T-PRoPHET in absence of malicious nodes, Fig. 14(c).

Fig. 15 plots Delivery Ratio of Wise-PRoPHET with varying percentage of selfish and malicious nodes. The plot provides us a design guideline to decide the maximum percentage of selfish and malicious nodes that can be tolerated to achieve a given level of Delivery Ratio. It is observed that our technique tolerates up to 30% selfish nodes and 40% malicious nodes to achieve a delivery ratio of 65%. Similar design guidelines can be drawn for achieving the desired level of Average Delay and Overhead Ratio.

6. Conclusion & future work

In this paper, a trust based Watchdog technique is proposed that

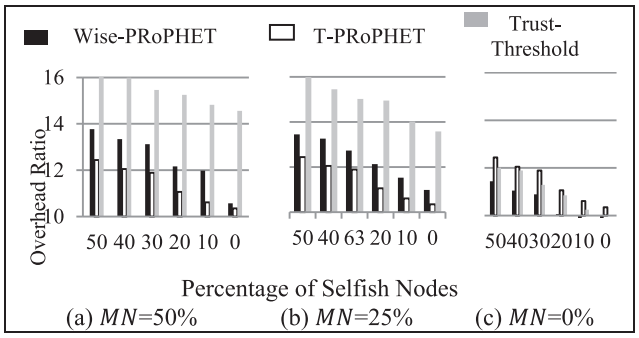

Fig. 14. Overhead Ratio with different levels of selfishness.

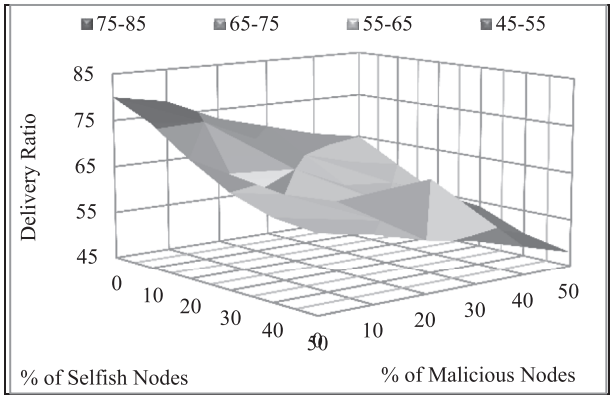

Fig. 15. Delivery Ratio of Wise-PRoPHET with different levels of selfishness and maliciousness.

monitors its neighbouring nodes to generate a local perception about their forwarding behaviour. An absorbing finite-state Markov chain is used to determine the expected message delays that assist a node in accurate monitoring. This local knowledge is then shared with other nodes aiming at generating a global perception for detecting selfish nodes in the network. The Watchdog also identifies the malicious nodes that collude to either spoil the reputation of a benign node or boost the reputation of a selfish node. The technique is further tuned to rationalize the self-trusting effect towards improving the delivery ratio as well as delay. Finally, the Watchdog is seamlessly integrated with PRoPHET routing protocol for on-the-fly detection and avoidance of selfish and malicious forwarders in a DTN based post disaster communication network. Overhead analyses are done to justify the application of Wise-PRoPHET in a resource constrained low-power smartphones based DTN. Simulation results justify that our technique outperforms a couple of state-of-the-art competing schemes. Wise-PRoPHET provides a design guideline to decide the percentage of selfish and malicious nodes that can be tolerated to achieve a given delivery ratio. In particular, it tolerates up to 30% selfish and 40% malicious nodes to deliver up to 65% situational messages in our application domain.

与prophet无缝结合

动态检测

However, in the proposed technique, there is a possibility that a node has indeed forwarded a message but the corresponding *FE* has not reached within the timeout period, due to the fragile network characteristics such as, network partitions. We aim to address this issue in future to make the proposed technique more rational.

References

Ayday, E., Lee, H., Fekri, F., 2010. Trust management and adversary detection for DTN. In: Proceedings of MILCOM (CA, USA, October 31-November 2, 2010).

Basu, S., Bhattacharjee, S., Roy, S., Bandyopadhyay, S., 2015. Sage-PRoPHET: a security aided and group encounter based PRoPHET routing protocol for dissemination of post disaster situational data. In: Proceedings of ICDCN (Goa, India, January 4-7, 2015).

Bhattacharjee, S., Roy, S., Bandyopadhyay, S., 2014. Exploring an energy-efficient DTN framework supporting disaster management services in post disaster relief operation. Wirel. Netw. 21 (3), 1033–1046 (Oct. 2014).

Burgess, J., Gallagher, B., Jensen, D., Levine, B.N., 2006. Maxprop: routing for vehicle-based disruption-tolerant networking. In: Proceedings of INFOCOM (Barcelona, Spain, April 23-26, 2006.

Campillo, A.M., Crowcroft, J., Yoneki, E., Martı, R., 2013. Evaluating opportunistic networks in disaster scenarios. J. Netw. Comput. Appl. 36 (2013), 870–880.

Chen, I.R., Bao, F., Chang, M., Cho, J.H., 2011. Integrated social and QoS trust-based routing in delay tolerant networks. Wireless Pers. Commun. 66 (2), 443–459 (Sep. 2012).

Chen, I.R., Bao, F., Chang, M., Cho, J.H., 2013. Dynamic trust management for DTN and its application in secure routing. IEEE Trans. Parallel Distr. Syst. 25 (5), 1200–1210 (May. 2014).

Chenji, H., et al., 2011. A Wireless Sensor, AdHoc and Delay Tolerant Network System for Disaster Response. Technical Report. LENSS-09–02.

Cho, Y., Qu, G., Wu, Y., 2012. Insider threats against trust mechanism with watchdog and defending approaches in WSNs. In: Proceedings of SPW (San Francisco, CA, USA, May 24-25, 2012).

Dias, J.A.F.F., Rodrigues, J.J.P.C., Xia, F., Mavromoustakis, C.X., 2015. A cooperative watchdog system to detect misbehavior nodes in vehicular delay-tolerant networks. IEEE Trans. Ind. Electron. 62 (12), 7929–7937 (Dec. 2015).

Dini, G., Duca, A.L., 2012. Towards a reputation-based routing protocol to contrast blackholes in a delay tolerant network. Ad Hoc Netw. 10 (7), 1167–1178 (Sep. 2012).

Doctors For You. http://d Doctors For You. http://doctorsforyou.org.

Fall, K., et al., 2010. A disruption-tolerant architecture for secure and efficient disaster response communications. In: Proceedings of ISCRAM (Washington, USA, May 2-5, 2010.

Groenevelt, R., Nain, P., Koole, G., 2005. The message delay in mobile adhoc networks. Perform. Eval 62 (1–4), 210–228 (2005).

Guo, X.F., Chan, M.C., 2013. Plankton: an efficient DTN routing algorithm. In: Proceedings of SECON (New Orleans, LA, USA, June 24-27, 2013).

Haas, Z., Small, T., 2006. A new networking model for biological applications of ad hoc sensor networks. IEEE/ACM Trans. Netw. 14 (1), 27–40 (2006).

International Federation of Red Cross and Red Crescent Societies, 2013. World Disasters Report 2013-Focus on Technology and the Future of Humanitarian Action. http://www.ifrc.org/PageFiles/134658/WDR%202013%20complete.pdf.

Jain, S., Demmer, M., Patra, R., Fall, K., 2005. Using redundancy to cope with failures in a delay tolerant network. In: Proceedings of SIGCOMM (Pennsylvania, USA, August 22-26, 2005.

Keranen, A., Ott, J., Karkkainen, T., 2009. The one simulator for DTN protocol evaluation. In: Proceedings of SIMUTools (Rome, Italy, March 2-6, 2009).

Lehmann, E.L., Casella, G., 1998. Theory of Point Estimation, second ed. Springer Texts in Statistics, California, USA.

Li, N., Das, S., 2013. A trust-based framework for data forwarding in opportunistic networks. Ad Hoc Netw. 11 (4), 1497–1509 (Jun. 2013).

Lindgren, A., Doria, A., Schelen, O., 2003. Probabilistic routing in intermittently connected networks. SIGMOBILE Mobile Comp. Commun. Rev. 7 (3), 19–20 (Jul. 2003).

Luo, H., Kravets, R., Abdelzaher, T., 2010. The-day-after Networks: a First-response Edge-network Architecture for Disaster Relief. NSF NeTS FIND Initiative.

Map of Water, Food, Shelter and Medical Resources, 2015. https://www.google.com/maps/d/viewer?mid=1Iv7GILViqyJAFn5o5hi1F2Fg8mc&hl=en_USUS. 2015.

Marmol, F.G., Perez, G.M., 2012. TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks. J. Netw. Comput. Appl. 35 (2012), 934–941.

Marti, S., Giuli, T.J., Lai, K., Baker, M., 2000. Mitigating routing misbehaviours in MANETs. In: Proceedings of MobiCom (Massachusetts, USA, August 06-11, 2000.

Ntareme, H., Zennaro, M., Pehrson, B., 2011. Delay Tolerant Network on smartphones: applications or communication challenged areas. In: Proceedings of ExtremeCom (Manaus, Brazil, September 26-30, 2011.

Orallo, E.H., Olmos, H., Cano, J.C., Calafate, C.T., Manzoni, P., 2015. CoCoWa: a collaborative contact-based watchdog for detecting selfish nodes. IEEE Trans. Mobile Comput. 14 (6), 1162–1175 (Jul. 2015).

Uddin, M.Y.S., Nicol, D.M., Abdelzaher, T.F., Kravets, R.H., 2009. A post-disaster mobility model for delay tolerant networking. In: Proceedings of WSC (TX, USA, December 13-16, 2009).

Vahdat, A., Becker, D., 2000. Epidemic Routing for Partially Connected AdHoc Networks. Technical Report. Department of Computer Science, Duke University.

Wang, W., Zwolinski, M., 2013. An improved instruction-level power model for ARM11 microprocessor. In: Proceedings of HIP3ES (Berlin, Germany, January 23, 2013).

Wang, B., Chen, X., Chang, W., 2014. A light-weight trust-based QoS routing algorithm for AdHoc Networks. Pervasive Mob. Comput. 13, 164–180 (Aug. 2014).

Zhou, P., et al., 2015. Toward energy-efficient trust system through watchdog optimization for WSNs – zhou, 2015. IEEE Trans. Inf. Forensics Secur. 10 (3), 613–625 (Mar. 2015).

Zhu, H., et al., 2014. A probabilistic misbehavior detection scheme towards efficient trust establishment in DTN. IEEE Trans. Parallel Distr. Syst. 25 (1), 22–32 (Jan. 2014).